UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/806,866	03/22/2004	Robert Tod Dimpsey	AUS920040065US1	2686

35525 7590 10/05/2006

IBM CORP (YA)
C/O YEE & ASSOCIATES PC
P.O. BOX 802333
DALLAS, TX 75380

EXAMINER	
SAVLA, ARPAN P	

ART UNIT	PAPER NUMBER
2185	

DATE MAILED: 10/05/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/806,866	DIMPSEY ET AL.	
	Examiner	Art Unit	
	Arpan P. Savla	2185	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *22 March 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-23* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-23* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *22 March 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *See Continuation Sheet*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

Continuation of Attachment(s) 3). Information Disclosure Statement(s) (PTO/SB/08), Paper No(s)/Mail Date :6/29/05, 3/27/06, 4/25/06, 5/31/06, 8/29/06.

DETAILED ACTION

The instant application having Application No. 10/806,866 has a total of 23 claims pending in the application, there are 4 independent claims and 19 dependent claims, all of which are ready for examination by the Examiner.

INFORMATION CONCERNING OATH/DECLARATION

Oath/Declaration

1. Applicant's oath/declaration has been reviewed by Examiner and is found to conform to the requirements prescribed in 37 CFR 1.63.

INFORMATION CONCERNING DRAWINGS

Drawings

2. Applicant's drawings submitted March 22, 2004 are acceptable for examination purposes.

ACKNOWLEDGMENT OF REFERENCES CITED BY APPLICANT

Information Disclosure Statement

3. As required by MPEP § 609(c), Applicant's submission of the Information Disclosure Statements dated June 29, 2005, March 27, 2006, April 25, 2006, May 31, 2006, and August 29, 2006 are acknowledged by Examiner and cited references have been considered in the examination of the claims now pending. As required by MPEP §

609 c(2), a copy of the PTOL-1449 initialed and dated by Examiner is attached to the instant office action.

4. Reference BJ on the Information Disclosure Statement dated June 29, 2005 has **not** been considered by Examiner because the reference does not comply with 37 CFR 1.98 (a)(3)(ii). Reference BJ is a non-English-language document, however, there is no copy of an English-language translation of the document, or portion thereof. Applicant is advised that the date of any re-submission of any item of information contained in this information disclosure statement or the submission of any missing element(s) will be the date of submission for purposes of determining compliance with the requirements based on the time of filing the statement, including all certification requirements for statements under 37 CFR 1.97(e). See MPEP § 609.05(a).

REJECTIONS NOT BASED ON PRIOR ART

Double Patenting

The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to

be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

5. **Claims 1-3, 6-7, 9, 12-14, 17-20, and 23** are provisionally rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over **claims 1, 3-6, 8, 11, 13-15, 18, and 20-22** of copending Application No. 10/806,871. Although the conflicting claims are not identical, they are not patentably distinct from each other because claims 1-3, 6-7, 9, 12-14, 17-20, and 23 of the instant application are covered by claims 1, 3-6, 8, 11, 13-15, 18, and 20-22 of copending application.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

Claims 1-3, 6-7, 9, 12-14, 17-20, and 23 of the instant application are compared to claims 1, 3-6, 8, 11, 13-15, 18, and 20-22 of the co-pending application below:

Instant Application 10/806,866	**Co-pending Application 10/806,871**
1. A method in a data processing system for providing hardware assistance to prefetch data during execution of code by a process or in the data processing system, the method comprising: responsive to loading of an instruction in the code into a cache, determining, by a	1. A method in a data processing system for providing hardware assistance to prefetch data during execution of code by a processor in the data processing system, the method comprising: responsive to loading an instruction in the code into a cache, determining, by a processor unit,

processor unit, whether **metadata for a prefetch is present for the instruction**; and responsive **to metadata being present for** the instruction, selectively prefetching **data, from within a data structure using the metadata,** into the cache in a processor.	whether **a prefetch indicator is associated with the instruction**; and responsive to **the prefetch indicator being associated with** the instruction, selectively prefetching **a pointer to a data structure identified by the prefetch indicator** into the cache in the processor.
2. The method of claim 1, wherein the selectively prefetching step **comprises**: determining whether outstanding cache misses are present; and prefetching the data if a number of outstanding cache misses are less than a threshold.	3. The method of claim 1, wherein the selectively prefetching step **includes**: determining whether outstanding cache misses are present; and prefetching the data if a number of outstanding cache misses are less than a threshold.
3. The method of claim 1, wherein the selectively prefetching step includes: determining whether to replace cache lines; and prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold.	4. The method of claim 1, wherein the selectively prefetching step includes: determining whether to replace cache lines; and prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold.
6. The method of claim 1, wherein the processor unit is selected from one of an instruction cache or a load/store unit.	5. The method of claim 1, wherein the processor unit is selected from one of an instruction cache, **data cache,** or a

	load/store unit.
7. The method of claim 1, wherein the cache is an instruction cache.	6. The method of claim 1, wherein the cache is an instruction cache.
9. A data processing system comprising: a cache in a processor in the data processing system; and a load/store unit in the processor, wherein the load/store unit determines **whether metadata for a prefetch is present** in response to loading an instruction for execution into a cache, the load/store unit selectively prefetches **data from within a data structure** into the cache using the metadata associated with the instruction.	8. A data processing system comprising: a cache in a processor in the data processing system; and a load/store unit in the processor, wherein the load/store unit determines whether **a prefetch indicator is associated with an instruction** in response to loading the instruction for execution into the cache, the load/store unit selectively prefetches **a pointer to a data structure identified by the prefetch indicator** into the cache using metadata associated with the instruction.
12. A data processing system for providing hardware assistance to prefetch data during execution of code by a process or in the data processing system, the data processing system comprising: determining means, responsive to loading of an instruction in the code into a cache,	11. A data processing system for providing hardware assistance to prefetch data during execution of code by a process or in the data processing system, the data processing system comprising: determining means, responsive to loading an instruction in the code into a cache, for

for determining, by a processor unit, **whether metadata for a prefetch is present for** the instruction; and selectively prefetching means, responsive to **metadata being present for the instruction**, for selectively prefetching **data, from within a data structure using the metadata,** into the cache in a processor.	determining, by the a processor unit, whether **a prefetch indicator is associated with** the instruction; and selectively prefetching means, responsive to **the prefetch indicator being associated with the instruction**, for selectively prefetching **a pointer to a data structure identified by the prefetch indicator** into the cache in the processor.
13. The data processing system of claim 12, wherein the selectively prefetching means **comprises**: **first** means for determining whether outstanding cache misses are present; and **second** means for prefetching the data if a number of outstanding cache misses are less than a threshold.	13. The data processing system of claim 11, wherein the selectively prefetching means **includes**: means for determining whether outstanding cache misses are present; and means for prefetching the data if a number of outstanding cache misses are less than a threshold.
14. The data processing system of claim 12, wherein the selectively prefetching means includes: **first** means for determining whether to replace cache lines; and **second** means for prefetching	14. The data processing system of claim 11, wherein the selectively prefetching means includes: means for determining whether to replace cache lines; and means for prefetching the data if a number of

the data if a number of cache lines chosen to be replaced are greater than a threshold.	cache lines chosen to be replaced are greater than a threshold.
17. The data processing system of claim 12, wherein the processor unit is selected from one of an instruction cache or a load/store unit.	15. The data processing system of claim 11, wherein the processor unit is selected from one of an instruction cache, **a data cache,** or a load/store unit.
18. A computer program product in a computer readable medium for providing hardware assistance to prefetch data during execution of code by a process or in the data processing system, the computer program product comprising: first instructions, responsive to loading of an instruction in the code into a cache, for determining, by a processor unit, whether **metadata for a prefetch is present for** the instruction; and second instructions, responsive to **metadata being present for** the instruction, for selectively prefetching **data, from within a data structure using the metadata,** into the	18. A computer program product in a computer readable medium for providing hardware assistance to prefetch data during execution of code by a process or in the data processing system, the computer program product comprising: first instructions, responsive to loading an instruction in the code into a cache, for determining, by the a processor unit, whether **a prefetch indicator is associated with** the instruction; and second instructions, responsive to **the prefetch indicator being associated with** the instruction, for selectively prefetching **a pointer to a data structure**

cache in a processor.	**identified by the prefetch indicator** into the cache in the processor.
19. The computer program product of claim 18, wherein the second instructions **comprises**: first sub-instructions for determining whether outstanding cache misses are present; and second sub-instructions for prefetching the data if a number of outstanding cache misses are less than a threshold.	20. The computer program product of claim 18, wherein the second instructions **includes**: first sub-instructions for determining whether outstanding cache misses are present; and second sub-instructions for prefetching the data if a number of outstanding cache misses are less than a threshold.
20. The computer program product of claim 18, wherein the second instructions includes: first sub-instructions for determining whether to replace cache lines; and second sub-instructions for prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold.	21. The computer program product of claim 18, wherein the second instructions includes: first sub-instructions for determining whether to replace cache lines; and second sub-instructions for prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold.
23. The computer program product of claim 18, wherein the processor unit is selected from one of an instruction cache or a load/store unit.	22. The computer program product of claim 18, wherein the processor unit is selected from one of an instruction cache, **a data cache,** or a load/store unit.

6. **As per claims 1, 9, 12, and 18** in the instant application and **claims 1, 8, 11, and 18** in co-pending application 10/806,871, one difference between the claims are whether metadata is present for the instruction as opposed to whether a prefetch indicator is associated with the instruction

However, at the time of the invention it would have been obvious to a person of ordinary skill in the art to implement the specific prefetch indicator more broadly as simply metadata.

Another difference between the claims is prefetching data from within a data structure using the metadata as opposed to prefetching a pointer to a data structure identified by the prefetch indicator.

However at the time of the invention it would have been obvious to a person of ordinary skill in the art to implement the specific pointer to a data structure more broadly as simply data.

The motivation for doing so would have been to improve performance due to reducing the time spent waiting for memory accesses to complete by pointing to a data location that is likely to be referenced.

Therefore, it would have been obvious to implement the prefetch indicator and pointer to a data structure respectively of claims 1, 8, 11, and 18 of co-pending application 10/806,871 as the metadata and data of claims 1, 9, 12, and 18 in the instant application.

Claim Rejections - 35 USC § 101

7. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

8. **Claims 18-23 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.** Claims 18-23 are not limited to tangible embodiments. In view of Applicant's disclosure, pg. 59, paragraph 1, the computer readable medium is not limited to tangible embodiments, instead being define as including both tangible embodiments (e.g. recordable-type media, such as a floppy disk, hard disk drive, a RAM, CD-ROMS, and DVD-ROMS) and intangible embodiments (e.g. transmission-type media, such as digital and analog communications links, wired or wireless communications links using transmission forms, such as, for example, radio frequency and light wave transmissions). As such, claims 18-23 are not limited to statutory subject matter and are therefore non-statutory.

REJECTIONS BASED ON PRIOR ART

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

10. Claims 1, 4-12, 15-18, and 21-23 are rejected under U.S.C. 102(e) as being anticipated by Damron (U.S. Patent 6,782,454).

11. **As per claims 1 and 18**, Damron discloses a method in a data processing system for providing hardware assistance to prefetch data during execution of code by a process or in the data processing system, the method comprising:

responsive to loading of an instruction in the code into a cache, determining, by a processor unit, whether metadata for a prefetch is present for the instruction (col. 4, lines 58-61; col. 5, lines 17-22; Fig. 1, element 175; Fig. 3, element 220); *It should be noted that computer program product in claims 18-23 executes the exact same functions as the methods in claims 1-6. Therefore, any references that teach claims 1-6 also teach the corresponding claims 18-23. It should also be noted that the "prefetch request" is analogous to the "instruction", the "prefetch engine" is analogous to the "processing unit", and the "starting address of a node, an offset value, and a termination value" all are analogous to the "metadata."*

and responsive to metadata being present for the instruction, selectively prefetching data, from within a data structure using the metadata, into the cache in a processor (col. 5, lines 24-26; Fig. 3, element 225).

12. **As per claims 4 and 21**, Damron discloses retrieving the data from within the data structure using a pointer and an offset value (col. 5, lines 20-21). *It should be noted that the "starting address of a node" is analogous to a "pointer."*

13. **As per claims 5 and 22**, Damron retrieving the data from the data structure using an address (col. 5, lines 20-21).

14. **As per claims 6 and 23**, Damron discloses the processor unit is selected from one of an instruction cache or a load/store unit (col. 4, lines 58-61; Fig. 1, element 175). *It should be noted that the "prefetch engine" is analogous to a "load/store unit."*

15. **As per claim 7**, Damron discloses the cache is an instruction cache (col. 4, lines 53-54).

16. **As per claim 8**, Damron discloses the metadata includes the pointer and the offset value (col. 5, lines 20-21). *See the citation note for claims 4 and 21 above.*

17. **As per claim 9**, Damron discloses data processing system comprising:

a cache in a processor in the data processing system (col. 4, lines 45-48);

and a load/store unit in the processor, wherein the load/store unit determines whether metadata for a prefetch is present in response to loading an instruction for execution into a cache, the load/store unit selectively prefetches data from within a data structure into the cache using the metadata associated with the instruction (col. 4, lines 58-61; col. 5, lines 17-22 and 24-26; Fig. 1, element 175; Fig. 3, elements 220 and 225). *See the citation note for claims 1 and 18 above.*

18. **As per claim 10**, Damron discloses the metadata is an address to the data within the data structure (col. 5, lines 20-21).

19. **As per claim 11**, Damron discloses the metadata is a pointer and an offset to the data within the data structure (col. 5, lines 20-21). *See the citation note for claims 4 and 21 above.*

20. **As per claim 12**, Damron discloses a data processing system for providing hardware assistance to prefetch data during execution of code by a process or in the data processing system, the data processing system comprising:

determining means, responsive to loading of an instruction in the code into a cache, for determining, by a processor unit, whether metadata for a prefetch is present for the instruction (col. 4, lines 58-61; col. 5, lines 17-22; Fig. 1, element 175; Fig. 3, element 220); *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 1 and 18 above.*

and selectively prefetching means, responsive to metadata being present for the instruction, for selectively prefetching data, from within a data structure using the metadata, into the cache in a processor (col. 5, lines 24-26; Fig. 3, element 225). *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 1 and 18 above.*

21. **As per claim 15**, Damron discloses retrieving means for retrieving the data from within the data structure using a pointer and an offset value (col. 5, lines 20-21). *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 4 and 21 above.*

22. **As per claim 16**, Damron discloses retrieving means for retrieving the data from the data structure using an address (col. 5, lines 20-21). *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer.*

23. **As per claim 17**, Damron discloses the processor unit is selected from one of an instruction cache or a load/store unit (col. 4, lines 58-61; Fig. 1, element 175). *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 6 and 23 above.*

Claim Rejections - 35 USC § 103

24. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

25. Claims 2, 13, and 19 are rejected under 35 U.S.C. 103(a) as being obvious over Damron in view of IBM Technical Disclosure, "Cache Miss Director – A Means of Prefetching Cache Missed Lines," hereafter "IBMTD."

26. **As per claims 2 and 19**, Damron discloses all the limitations of claims 2 and 19 except determining whether to replace cache lines;

and prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold.

IBMTD discloses determining whether outstanding cache misses are present

(Discourse Text, lines 13-14); *It should be noted that the "demand miss" is analogous to the "cache miss."*

and prefetching the data if a number of outstanding cache misses are less than a threshold (Discourse Text, lines 14-17). *It should be noted that this limitation contains language that suggests or makes optional but does not require steps to be performed or does not limit the claim to a particular structure and therefore does not limit the scope of a claim. The term 'if' denotes an optionally recited limitation and optionally recited limitations are not guaranteed to take place. Thus, simply "prefetching the data" is disclosed because the optionally recited parts of this limitation are not required to be taught by the Office. See MPEP §2106, Section II(C)). It should also be noted that "anticipatory cache misses" are analogous to "prefetching data."*

Damron and IBMTD are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement IBMTD's Cache Miss Directory (CMD) within Damron's system for prefetching data within pointer linked data structures.

The motivation for doing so would have been to get the lines into the cache before a demand miss occurs for them, thus, reducing processing delays (IBMTD, Discourse Text, lines 2-3 and 19-20).

Therefore, it would have been obvious to combine Damron and IBMTD for the benefit of obtaining the invention as specified in claims 2 and 19.

27. **As per claim 13**, the combination of Damron/IBMTD discloses first means for

determining whether outstanding cache misses are present (Discourse Text, lines 13-14); *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 3 and 19 above.*

and second means for prefetching the data if a number of outstanding cache misses are less than a threshold (Discourse Text, lines 14-17). *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 2 and 19 above.*

28. Claims 3, 14, and 20 are rejected under 35 U.S.C. 103(a) as being obvious over Damron in view of Malik (U.S. Patent 6,687,794).

29. **As per claims 3 and 20**, Damron discloses all the limitations of claims 3 and 20 except determining whether to replace cache lines;

and prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold.

Malik discloses determining whether to replace cache lines (col. 4, lines 60-63);

and prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold (col. 5, lines 16-19). *It should be noted that this limitation contains language that suggests or makes optional but does not require steps to be performed or does not limit the claim to a particular structure and therefore does not limit the scope of a claim. The term 'if' denotes an optionally recited limitation and optionally recited limitations are not guaranteed to take place. Thus, simply "prefetching the data" is disclosed because the optionally recited parts of this limitation are not*

required to be taught by the Office. See MPEP §2106, Section II(C)).

Damron and Malik are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Malik's prediction history within Damron's system for prefetching data within pointer linked data structures.

The motivation for doing so would have been to provide a technique to increase the performance of the data cache by reducing the possibility if a data cache miss (Malik, col. 4, lines 53-55).

Therefore, it would have been obvious to combine Damron and Malik for the benefit of obtaining the invention as specified in claims 3 and 20.

30. **As per claim 14**, the combination of Damron/Malik discloses first means for determining whether to replace cache lines (Malik, col. 4, lines 60-63); *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer.*

and second means for prefetching the data if a number of cache lines chosen to be replaced are greater than a threshold (Malik, col. 5, lines 16-19). *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 3 and 20 above.*

Conclusion

STATUS OF CLAIMS IN THE APPLICATION

The following is a summary of the treatment and status of all claims in the application as recommended by MPEP 707.70(i):

CLAIMS REJECTED IN THE APPLICATION

Per the instant office action, **claims 1-23** have received a first action on the merits and are subject of a first action non-final.

RELEVANT ART CITED BY THE EXAMINER

The following prior art made of record and not relied upon is cited to establish the level of skill in Applicant's art and those arts considered reasonably pertinent to Applicant's disclosure. See MPEP 707.05(e).

1. U.S. Patent 5,652,858 (Okada et al.) discloses a method for prefetching pointer-type data structures.

2. U.S. Patent 5,822,790 (Mehrotra) discloses a voting data prefetch engine.

3. U.S. Patent 6,687,807 (Damron) discloses a method and apparatus for prefetching linked data structures.

4. U.S. Patent 6,848,029 (Coldewey) discloses a Method and apparatus for prefetching recursive data structures.

5. Disclosed Anonymously, "Method for the dynamic prediction of nonsequential memory accesses", September 25, 2002, ip.com, IPCOM000009888D.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Arpan P. Savla whose telephone number is (571) 272-1077. The examiner can normally be reached on M-F 8:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Sanjiv Shah can be reached on (571) 272-4098. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Arpan Savla
Art Unit 2185
September 29, 2006

SANJIV SHAH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100